Properties

- Admissible / monotonic non-decreasing / consistent
- Admissible != monotonicity
- Admissible != consistent
- Admissible != first path opened to any node is optimal
- Consistent => admissible
- Consistent => nodes are not re-opened

**A***

- Best-First Search
  - $f = g + h$
  - $f$ is an estimate of the complete path length
- Optimality?
  - Not optimal unless we guarantee properties of our heuristic

**Optimality of A***

- If edges have minimum positive cost
- If $h$ is non-negative and non-overestimating
  - admissible
- The goal node will be expanded when we have an optimal path to the goal
A* Optimality Proof (1)

- Induction like Dijkstra’s to show that we always have a node on a path to the goal on the open list (with optimal cost)
- If there is a path to a goal we will eventually find it
  - Nodes on the path are always on OPEN
- If we run forever, there cannot be a path to a goal
  - $g$ is always increasing

A* Optimality Proof (2)

- When a goal is chosen for expansion, the cost must be optimal
  - Suppose goal has f-cost, $K$
  - All nodes in OPEN have cost $\geq K$
  - $h$ is a lower bound, so $f$ is a lower bound
  - No path to goal with cost less than $K$

A* - Analysis

- Time Complexity
  - How many nodes do we generate?
  - How expensive is the open list?
- Assume $h(n) = 0$ for all $n$
  - Same as Dijkstra $O(b^{c/e})$
- Assume $h(n) = h^*(n)$ for all $n$
  - Will go straight to the goal(???)

A* - Time Optimality (in node expansions)

- Assume consistent heuristic
- Compare to all algorithms which find opt. sol.
- For a given consistent heuristic, every admissible algorithm must expand all nodes expanded by $A^*$
  - admissible algorithm = guaranteed to find opt. solution
  - nodes surely expanded by $A^* = f(n) < c$
  - $c = \text{optimal solution cost}$
A* Optimality Proof (1)

- Assume there exists an algorithm B that expands less nodes that A* on problem P
  - There must be some node $m$ not expanded by B, but expanded by A*
    - $f(m) = g(m) + h(m) < c$
  - Create a new problem P' with new goal $g'$
  - Add an edge with cost $h(m)$ to $g'$

Can we make A* go faster?

- A*: $f(n) = g(n) + h(n)$
- Weighted A*: $F(n) = (1-w) \cdot g(n) + w \cdot h(n)$
  - If $w = 1$?
    - Pure Heuristic Search
  - If $w = 0$?
    - Dijkstra's

A* Optimality Proof (2)

- Goal $g'$ is now shorter than the original goal
  - $g(g') = g(m) + c(m, g') = g(m) + h(m) = f(m) < c$
  - $g'$ will not be found by B, but will be found by A*

  Is our heuristic still consistent/admissible?
  - $|h(m) - h(g')| \leq c(m, g')$
  - $|c(m, g') - 0| \leq c(m, g')$

Demonstrate A*

- Show A* on several problems
  - Expand one node at a time
  - Examine tie-breaking
  - Difference between Manhattan distance and octile distance
### IDA*
- Previously, BFS $\rightarrow$ DFS $\rightarrow$ DFID
- Now, A* $\rightarrow$ IDA*
- Perform DFS within f-cost limits
- Korf, 1985

### IDA* Pseudo-Code
```
IDA*(start, goal)
limit ← f-cost(start)
do
    path = cost-limited-DFS(start, goal, limit)
    limit ← newlimit
while (!path)
return path
```

### Cost limits
- How do we determine the next cost limit?
  - Keep track of the minimum f-cost larger than limit found during search
  - This is the next limit

### Example
- $[0 \ 1 \ 2 \ | \ 3 \ 4 \ 5]$
- $[0 \ 1 \ 2 \ | \ 4 \ 3 \ 5]$
- Admissible heuristic
Termination

- Assume
  - Solution exists
  - Edges costs have minimum finite value
  - Non-negative heuristic values

- If no heuristic, uniform edge-cost, same as DFID

Termination (2)

- All path costs are strictly increasing
- All nodes with a given cost are expanded in each iteration
  - Cost-limit strictly increasing
- At least 1 new node expanded each iteration
- No infinite-length paths of finite cost
- Must eventually expand the goal

Optimality

- Frontier -- nodes which have been generated but not expanded
  - Frontier always contains node on optimal path to goal
- Cost thresholds are monotonically increasing
- No thresholds > optimal path length
  - $f(n) < \text{optimal solution cost}$
- Goal has $f(n) = g(n)$ -- no shorter solution
  - Cannot run with a threshold > $g(\text{goal})$

Space Complexity

- Assume goal has cost $c$, minimum edge cost $e$
- Maximum depth of $c/e$ (+1 for expanding at this depth)
- $e$ is constant, so space is $O(c)$